

一个基于多种评判模式的在线评判系统

王桂平 刘 君 罗 宪 陈旺桥

重庆交通大学信息科学与工程学院 重庆 400074

摘 要 以编程能力为核心的软件开发能力是电子及信息类专业学生都必须掌握的基本能力。在线程序实践和程序设计类学科竞赛能培养学生的学习兴趣,提高学生的程序实践能力。在线评判系统在程序设计类课程的实践教学和程序设计学科竞赛的推广中发挥着重要的作用。近十几年来,各类程序设计学科竞赛在国内高校日益推广,参赛面和影响力越来越大。目前亟需一种能适应不同评判模式的在线评判系统。文中分析了在线评判系统现状,总结了单数据集、多数据集、带权重多数据集 3 种评判模式。设计与开发了一个基于多种评判模式的在线评判系统——MOJ。MOJ 系统能适应不同的程序设计竞赛,也能用于程序设计类课程的实践教学,在教学和竞赛中发挥着重要的作用。

关键词: 学科竞赛;程序实践能力;程序设计竞赛;在线评判系统;多种评判模式

中图法分类号 TP391.7

Online Judge System Based on Multiple Judgement Modes

WANG Gui-ping, LIU Jun, LUO Xian and CHEN Wang-qiao

College of Information Science and Engineering, Chongqing Jiaotong University, Chongqing 400074, China

Abstract Software development abilities including programming are the basic abilities for students from electronic and information majors. Online programming practice and programming contest can cultivate students' learning interests and improve students' programming practice abilities. Online Judge (OJ) systems play important roles in practical teaching of programming courses, as well as in promoting of programming contests. In recent years, various programming contests are increasingly being promoted in domestic universities of China. The extent and influence of these contests are increasing. There is an urgent requirement for an OJ system that can adapt to different judgement modes. This paper analyzes the present situation of OJ systems. It summarizes three judgement modes, i. e., single data set, multiple data sets, and weighted multiple data sets. It designs and develops an OJ System based on multiple judgement modes (MOJ), and implements these three judgement modes in MOJ. MOJ can be adopted to different programming contests, as well as the practical teaching of programming courses. MOJ plays important roles in teaching and contests.

Keywords Academic contests, Programming practical ability, Programming contests, Online judge system, Multiple judgement modes

1 引言

以编程能力为核心的软件开发能力是电子及信息类专业学生都必须掌握的基本能力。在大数据及人工智能上升为国家战略的背景下,这些基本能力尤为重要。为此,各高校针对这些专业都开设了程序设计类课程,采用 C, C++, Java, C#, Python 等语言讲授。

程序设计类课程早期的教学方法多偏重于某种编程语言的语法知识教学。近年来,很多高校探索出以程序设计思想和方法的培养为主,突出实践能力培养的教学方法^[1-3]。此外,这些课程传统的实践教学通常由老师布置题目、学生课后编程实现再提交给老师人工评判。这种实践方式评判不及时且主观性强,难以真正提高学生的实践能力,老师也难以及时检验学生的学习效果。近年来,在线程序实践作为一种新颖的程序设计实践形式逐渐被引入到程序设计类课程中^[2-3]。

在线程序实践是指由在线评判(Online Judge, OJ)系统提供题目,学生在线提交程序, OJ 系统实时评判并反馈评判结果。这些题目一般具有较强的趣味性和挑战性,评判过程和结果公正及时,因而能引起初学者的极大兴趣。

此外开展程序设计学科竞赛是培养学生学习兴趣、提高学生程序实践能力的一种积极方式。这类学科竞赛具有参赛门槛比较低(只要学习一门编程语言就可以参赛)、趣味性高、层次多(国际级、国家级、区域级、省级、校级)、参赛机会多等特点,同时也具有较高的挑战性,所以吸引了大量学生参赛。近年来,各类程序设计学科竞赛在国内高校日益推广,参赛面和影响力越来越大。学科竞赛的激励、竞争和创新机制也弥补了传统实践教学在创新型人才培养上的不足,因此高校可以依托程序设计学科竞赛构建电子及信息类专业创新型人才培养体系^[4-5]。

OJ 系统在程序设计类课程的实践教学和程序设计学科

基金项目:重庆市高等教育教学改革研究重大项目(171016)

This work was supported by the Major Project of Teaching Reform and Research in Chongqing Higher Education (171016).

通信作者:王桂平(wgp@cqjtu.edu.cn)

竞赛的推广中发挥着重要的作用。目前亟需一种能适应不同评判模式的 OJ 系统。本文设计与开发了一个基于多种评判模式的在线评判系统(Online Judge with Multiple judgement modes, MOJ),实现了单数据集、多数据集、带权重多数据集 3 种评判模式。MOJ 可用于程序设计类课程的课后练习和上机考核、程序设计学科竞赛的开展等,具有广泛的意义和实用价值。

2 在线评判系统的现状分析

目前,国内高校广泛开展的程序设计学科竞赛主要有以下几类:

- 1)各省市大学生程序设计大赛、中国大学生程序设计大赛(China Collegiate Programming Contest,CCPC)、国际大学生程序设计大赛(ACM International Collegiate Programming Contest,ACM/ICPC)^[6],这些竞赛往往采用相似的竞赛规则和相同的评判模式。
- 2)蓝桥杯全国软件和信息技术专业人才大赛(以下简称“蓝桥杯大赛”),由国家工业和信息化部人才交流中心举办,包括省赛和全国总决赛,各省市一等奖进入全国总决赛。
- 3)团体程序设计天梯赛,是中国高校计算机大赛的 3 个竞赛版块之一,由教育部高等学校计算机类专业教学指导委员会等联合主办。

这些竞赛的评判主要有以下两种方式。

1)C/S(Client/Server)方式:主要指采用 PC2(Programming Contest Control System)^[7] 软件搭建的竞赛和评测环境。竞赛环境由服务器、裁判端、学生端等组成。学生端提交解答程序到服务器,服务器请求裁判端进行评判,评判结果返回到服务器端后,服务器实时反馈给学生端。这种方式局限性比较大:所有机器均需拷贝和配置 PC2 软件;服务器一般无法存储和显示题目,题目只能以其他方式(如纸质)提供给学生;每次竞赛都需配置环境并导入题目的数据文件等。

2)B/S(Browser/Server)方式(如文献[8]):指由 OJ 网站提供题目(OJ 网站一般会提供数以千计的极具趣味性和挑战性的题目),学生通过浏览器浏览题目,在线提交解答程序,OJ 网站的服务器实时评判并把结果反馈给用户。

为了评判用户提交程序的正确性,在线评判主要采用单数据集、多数据集、带权重多数据集 3 种评判模式。

1)单数据集:大学生程序设计大赛主要采用这种评判模式,每道题目有一个很大的数据集(即一个数据文件,但包含多组测试数据,可以多达上万组),用户程序需要通过整个数据集的评判才算正确。近几年,很多高级别的竞赛由于一道题目的测试数据太多,往往不得不拆分成多个数据文件,但每个数据文件仍然包含多组测试数据。这种评判模式应用较广,但要求非常苛刻,容易打击学生的积极性。

2)多数据集:每道题目对应若干个小的数据集(每个数据集都对应一个数据文件),用户的程序通过每个程序集的评判都会得到相应的分数(通常每个数据集的得分是一样的)。蓝桥杯大赛采用这种评判模式,蓝桥杯大赛官方提供的练习系统收录了数百道练习题,部分题目仅开放给 VIP 用户。

3)带权重多数据集:考虑到每个数据集的难度可能不一样,理应为每个数据集分配分数权重,简单的数据集分数权重

低,较难的数据集分数权重高。这种评判模式很少采用,但出于课程考核和鼓励积极提交程序的需要,这种评判模式是有积极意义的。

目前,单数据集评判模式应用较广,多数据集评判模式应用较少,带权重多数据集鲜有研究。本文系统地研究了这 3 种评判模式的特点,并在 MOJ 系统中实现了这 3 种评判模式。

由于 OJ 系统在程序设计类课程实践教学和程序设计竞赛推广方面的显著作用,计算机科学领域权威 SCI 期刊 ACM Computing Surveys(2018 年的影响因子为 5.55)在 2018 年 4 月刊出了一篇综述论文(见文献[9]),评述了目前 OJ 系统的现状。在 OJ 系统应用方面,Alemán 介绍了有关学生如何在 CS2 课程中使用 OJ 系统,以及该系统的排名和及时反馈对学生产生的强烈的积极影响^[10]。文献[11-13]分别介绍了 OJ 系统在 C++ 程序设计、数据结构等程序设计类课程中的应用,并提出了一些新颖的教学模式。文献[14]结合属性计数算法和 Rabin-Karp 算法对用户提交的源码进行比对,能有效地检测出源码的抄袭。

随着云计算技术的发展,一些 OJ 系统也部署或迁移到开放式云平台上^[15]。文献[16]结合 Docker 虚拟化技术和非对称分布式技术实现了一个分布式的程序评判系统。

3 MOJ 系统的架构

本文设计的 MOJ 系统架构如图 1 所示。MOJ 采用前后端分离的架构,后端采用 RESTful(Representational State Transfer,表现层状态转化)风格的应用程序接口(Application Programming Interface,API),这样能使各个平台(如 iOS,Android 和 Web)方便地调用服务接口。因此无须为不同的平台进行适配接口,大大提高了开发效率,降低了测试成本,且减少了系统的维护成本。

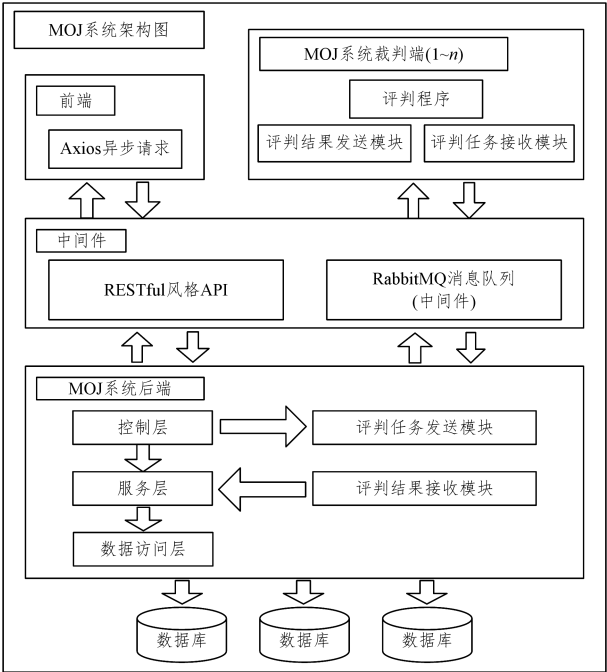


图 1 MOJ 系统的架构图

Fig. 1 System architecture of MOJ

后台采用 Spring Boot 框架进行开发,无须配置繁琐的配置文件,可以更加专注于数据访问层(Data Access Object layer)、服务层(Service layer)和控制层(Controller layer),前端只须专注于用户体验即可。

前后端分离还有一个更重要的优势是,前后端开发可以异步进行,其前提是前期做好需求分析、数据库设计以及详细设计。

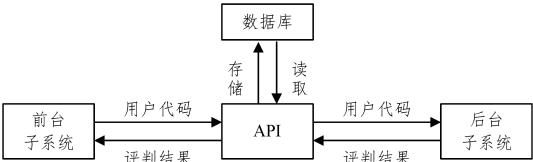
MOJ 系统使用 CentOS7.0 作为服务器的操作系统,用 Tomcat 作为 Web 服务器,其中 Tomcat 内嵌于 Spring Boot 框架中,采用 Oauth 2.0 框架用于身份认证,使用 MariaDB 作为后端的数据库,使用 Redis 作为服务器缓存,利用 Java 作为后端应用程序接口的主要开发语言。门户端和管理端采用前后端分离架构进行协同,使用 Vuejs 框架进行页面编写,页面将直接使用 Nginx 进行部署。

后端开发的 IDE 采用 Spring 官方推荐的 Spring Tool Suite(STS),并利用其开发 Spring Boot 项目,包括后端的 API 以及评判端的开发,同时使用 Gradle 作为 MOJ 项目的构建工具。页面的编写使用 Visual Studio Code 作为首选的 IDE,因为它是开源的、免费的且非常完美地支持 Vue 框架。同时采用 Git 作为 MOJ 项目的代码管理工具,用 Github 作为代码托管平台,方便整个团队协作开发一个系统。

4 MOJ 系统多种评判模式的实现

4.1 评判功能的体系结构

MOJ 系统的核心功能是对用户在线提交的程序代码进行实时的评判(包括编译、运行和比对),并将评判结果返回给用户,如图 2 所示。MOJ 分为前台子系统和后台子系统。前台子系统接收用户提交的代码,并存入数据库。

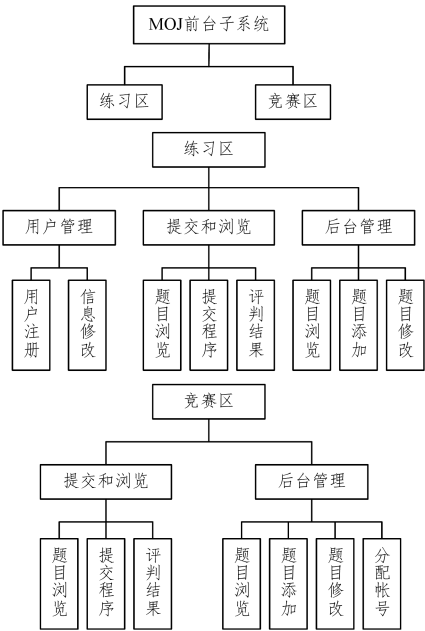


为了使 MOJ 系统既能适应程序设计竞赛,也能用于学生平时练习和程序设计类课程的实践教学,将 MOJ 的前台子系统分成两个不同功能模块:练习区和竞赛区,如图 3 所示。其中练习区允许用户注册和修改信息,但在竞赛区参赛队伍的帐号是人为分配的。

MOJ 后台子系统负责对用户提交的程序代码进行评判,并将评判结果写入数据库。根据用户的不同需要,前台子系统从数据库取出相应的数据回显于 Web 页面,供用户查看。在整个过程中,MOJ 系统的 API 类似于交通枢纽,只有 API 与数据库进行交互,而其他终端只需要通过中间件或者直接和 API 进行交互,这样数据流向变得更加清晰明了,不易出错。

MOJ 系统的 API 是整个在线评判系统的核心枢纽,类似于人脑,所有交互都依靠 API 进行转换及路由,因此 API 的设计至关重要。MOJ 系统后端采用 RESTful 风格的 API,这

样可以使得平台无关化,任何一个平台只需要调用 API 即可,不需要为不同的平台开发不同的接口,大大降低了开发成本以及速度。



4.2 多种评判模式

综合分析单数据集、多数据集和带权重多数据集这 3 种评判模式后发现,在实现时这 3 种评判模式可以合并成为一种,即单数据集和多数据集都可以看成带权重多数据集的特例,如表 1 所列。

表 1 多种评判模式的实现方式

Table 1 Implementation of multiple judgement modes		
	数据集个数 n	权重 m_i
单数据集	$n=1$	$m_1=100$
多数据集	$n(n>1)$	$m_i=m, \sum m_i=100$
带权重多数据集	$n(n>1)$	$1<m_i\leq 100, \sum m_i=100$

假设有 n 个数据集,每个数据集的权重为 $m_i(1\leq m_i\leq 100,1\leq i\leq n)$ 。单数据集可以看作只有一个数据集的带权重多数据集,即 $n=1, m_1=100$ 。多数据集可以看作有多个数据集但每个数据集的权重相同的带权重多数据集,即 $n>1, m_i=m, \sum m_i=n\times m=100$ 。而对带权重多数据集,则有 $n>1, 1\leq m_i\leq 100, \sum m_i=100$ 。约定所有数据集权重之和为 100,这样 3 种评判模式即可融合成为带权重多数据集,方便程序的实现。

在 MOJ 系统中,用户在添加每道题时,可以设置数据集个数 n 和每个数据集的权重 m_i ,从而实现支持多种评判模式。

4.3 MOJ 系统评判流程

MOJ 系统的后台子系统运行守护程序,并且监听消息队列,检查是否有新的提交记录。若存在新的提交记录,则启动评判程序进行评判。每道题目均对应相应的标准输入数据文件和标准输出数据文件。MOJ 系统的评判流程如图 4 所示。

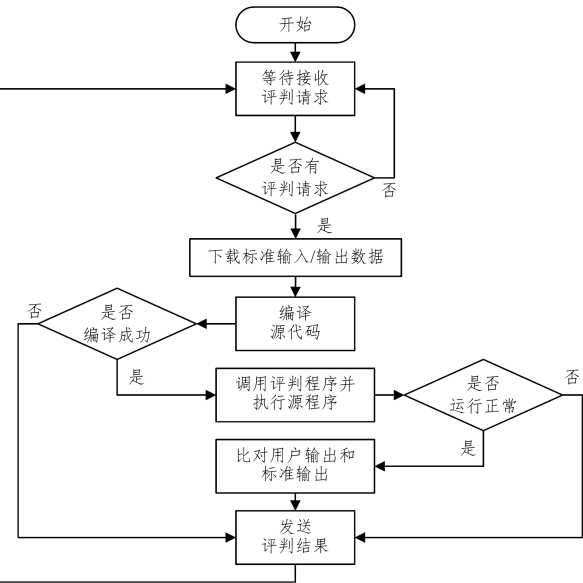


图 4 MOJ 系统评判流程
Fig. 4 Process of MOJ system judge ment

后端 API 接收到用户提交的源代码后,存储至对象存储服务(Minio)中,将提交信息(包括源代码的相对路径)、评判语言信息和测试数据信息通过消息队列(RabbitMQ)按照平均分配的策略分发给裁判端。裁判端有一个消息接收进程一直等待消息的到来,为保证评判的公平性,每一个裁判端一次只能接收一个消息并进行处理,直到处理完成后才进行下一次的接收,否则消息将会被积压至消息队列中。

裁判端接收到评判所需的各种元数据后,从对象存储服务上下载源代码和标准输入/输出数据文件,并保存至操作系统的临时目录下。首先对源代码进行编译,如果编译无误才进行下一步的执行;如果执行无误,则进行下一步的正确性对比。编译过程和执行过程都会受到时间限制、内存使用限制和安全限制等。最后评判程序将会得出结果,包括:评判通过(Accepted, AC)、编译错误(Compile Error, CE)、运行时异常(Runtime Error, RE)、超出时间限制(Time Limit Exceeded, TLE)、超出内存限制(Memory Limit Exceeded, MLE)和答案错误(Wrong Answer, WA)等。获得评判结果之后,将计算最后的得分并保存至评判信息中。同时评判程序将所有测试数据的评判信息都返回给 API,测试数据的评判信息包括评判结果、运行时间、运行内存等。API 将评判结果信息存入数据库,返回提交 ID 给客户端。这样,客户端就可以根据 ID 读取评判信息,回显于 Web 页面。

MOJ 系统的评判功能可以分解为以下 6 个模块:接收元数据、准备评判数据、编译源代码、运行源程序、对比结果和返回评判结果,具体阐述如下。

(1)接收元数据模块负责从消息队列中获取提交任务,接收评判所需要的元数据。该元数据包括初始提交信息(包括评判的时间限制和内存限制)、评判程序所在服务器的相对路径、源代码所在服务器的相对路径、评判语言的相关信息(包括编译命令模板、执行命令模板、评判所需操作系统、评判语言源文件扩展名和评判语言可执行文件扩展名等)。

(2)准备评判数据模块负责从对象服务器下载评判所需要的评判程序、需要评判的源代码、题目的标准输入/输出测试数据。根据评判语言的扩展名要求保存源文件,例如用

C++编写的源程序可能保存为 26c70347-f9de-41a9-b9e1-49deec39631c.cpp。特殊语言的源文件名可能进行特殊的处理。例如,对于 Java 语言,需要将源文件命名为 Main.java。

(3)编译源代码模块负责解析评判语言的编译命令模板,然后调用编译命令(如 gcc,g++或者 javac)对源文件进行编译。如果编译通过,则会生成可执行文件。当然,如果评判语言的编译命令模板为空,则不需要编译,说明该语言为解释型语言,默认编译通过。一旦编译过程中出现任何错误,将直接返回总体结果。

(4)运行源程序模块的功能是解析评判语言的运行命令模板,通过构建参数来调用评判程序进行评判。评判程序所需要的参数包括执行命令(--command_line,-C)、时间限制(--time_limit,-T)、内存限制(--memory_limit,-M)、标准输入文件路径(--input_file_path,-I)、程序输出文件路径(--output_file_path,-O)。例如:“./judge-linux-C “case/1001”-T 1000-M 32767-I case/input1.i-O case/output1.o”。在执行完成后,将获得程序的返回码(exitCode)、运行时间(usedTime)和运行内存(usedMemory)。根据 exitCode 判断用户程序的执行情况。如果 exitCode=0,则表示执行正常,未发生实质性的错误;如果 exitCode=1001,则表示超出内存限制;如果 exitCode=1010,则表示超出时间限制;如果 exitCode 为其他值,则表示运行时异常。

(5)对比结果模块主要负责将产生的输出结果和标准输出数据进行对比,其会自动过滤某些格式错误,比如换行符或者多余的空格。如果对比的结果有任何一处不一致,将会返回结果错误(Wrong Answer)。如果结果完全一致,则认为这个程序是正确的。不论正确与否,该模块将对此次评判进行自动打分,并封装提交信息。

(6)返回评判结果模块发送封装好的提交信息给 API,API 接收到信息之后将结果存储至数据库,最后将提交的 ID 返回给用户,用户可根据此 ID 查询评判结果。

4.4 测试及上线运行

MOJ 系统的核心功能就是接收用户提交的源代码、评判、返回评判结果,为了测试系统的可靠性、稳定性、并发性,分别提交正确的代码、有编译错误代码、有运行时异常代码、运行超时的代码、超内存限制的代码至 MOJ 系统,均能得到正确的反馈信息。

另外,MOJ 已经上线运行,并在大学生程序设计竞赛和蓝桥杯大赛的练习赛和校内选拔赛等一系列比赛中发挥了重要的作用。因此,MOJ 系统的应用前景非常广阔。

结束语 本文设计的在线评判系统——MOJ,实现了单数据集、多数据集、带权重多数据集 3 种评判模式,可以适应不同的程序设计竞赛,也能用于程序设计类课程的实践教学,在提升学生学习兴趣、激发编程激情、提高实践能力等方面发挥着极其重要的作用。

参 考 文 献

[1] HAN J P,LIU C Y,HU W H. A Programming Teaching Mode Based on Learning in-Class and out-Class and the Combination of Competitions with Classroom Teaching [J]. Research and Exploration in Laboratory,2014(6):169-171.
[2] WANG G P ,FENG R. Teaching methods focusing on cultivating practical ability in course of programming [J]. Laboratory

Science,2009(1):81-84.

[3] WANG G P,CHEN S Y,YANG X,et al. OJPOT:Online Judge & Practice Oriented Teaching Idea in Programming Courses [J]. European Journal of Engineering Education, 2016, 41 (3): 304-319.

[4] LIANG B ,FENG L. Teaching and practice methods of computer talents cultivation based on ACM/ICPC [J]. Laboratory Science, 2017(4): 229-231.

[5] ZHANG J P ,ZHANG S J. Subject Competition Promotes the Development of College Students' Educational Technology Ability [J]. E-education Research, 2010(8): 35-41.

[6] JU S G, LIAO Y, ZHOU G, et al. ACM contest and the Experimental Teaching Innovation [J]. Experimental Technology and Management, 2009, 26(5): 125-126.

[7] PC2[OL]. <http://pc2.ecs.csus.edu/>.

[8] QIAO S J, YANG Y, GE Y M, et al. The Design and Implement of Multi-user Online Judge System Based on B/S Mode [J]. Computer Engineering & Science, 2011, 33(A1): 58-61.

[9] WASIK S, ANTCHAK M, BADURA J, et al. A Survey on Online Judge Systems and Their Applications[J]. ACM Computing Surveys, 2018, 41(1): 3.

[10] ALEMÁN J L. Automated Assessment in a Programming Tools Course [J]. IEEE Transactions on Education, 2011, 54(4): 576-581.

[11] WANG L F, LI X Z, LIU Y K, et al. Application of Online Eva-

luation in Data Structure Practice Teaching [J]. Computer Education, 2016(4): 71-73.

[12] WANG R Z. Design of Mixed Learning Mode for Programming Courses Based on Online Evaluation System [J]. Computer Education, 2018(8): 126-129.

[13] WU S X , LIANG B L. Application of Online Judge in Practice Teaching of C++ Programming Course [J]. Journal of Higher Education, 2016(9): 164-165.

[14] CHEN R Q, HU Y L, Ying J J, et al. Study on similarity detection of source code in online judge system [J]. Experimental Technology and Management, 2014, 31(4): 109-111.

[15] ZHANG H B. Design and Implementation of the Open Cloud Platform Based Open Source Online Judge System [J]. Computer Science, 2012(11A): 339-343.

[16] SHU X F, ZHANG W, WANG M N. Design and Development of Distributed Program Judgment System Based on Docker [J]. Research and Exploration in Laboratory, 2018, 37(6): 140-143.



WANG Gui-ping, born in 1979, Ph. D, associate professor, M. S. supervisor, is a member of CCF. His main research interests include machine learning, big data and traffic infrastructure monitoring.

(上接第 650 页)

[13] XING Z Y. Wireless automatic Milk Tea Machine based on Mobile phone APP [J]. Scientific and Technological Innovation, 2019(6): 68-69.

[14] ZHU Z C, YAN J H, LIU C. Design of Pulse Wave Measurement system based on J2ME [J]. Electronic Design Engineering, 2011, 19(11): 71-75.

[15] CHEN J F, DENG M, TANG J L, et al. Research on the Communication Scheme of Real-time large-scale long-distance experiment [J]. Computer Engineering and Applications, 2018, 54(19): 94-100.

[16] LI Q D, WANG H, ZENG Y B, et al. Design and implementation of testing system for Wear Resistance of Aluminum Alloy Surface [J]. Equipment Manufacturing Technology, 2018(5): 99-102.

[17] KARLI W, CHRISTIAN N. Beginning Visual C# 2005[M]// Software Development, and C# Development. Beijing: Tsinghua University Press, 2006: 18-82.

[18] CHEN Y Z, LIN Q G, XIA J, et al. Mobile Monitoring System of Urban Air Quality Based on UAV[J]. Journal of Electronic Measurement and Instrument, 2019, 33(10): 1-6.

[19] SHEN C J, XING Z, ZHENG W G, et al. Design and implementation of remote Monitoring system for Agricultural products transaction based on Wi-Fi [J]. Journal of Chinese Agricultural Mechanization, 2012(4): 160-164.

[20] GUO L. The First Line of Code[M]// Computer, Software Development, and Android Development. Beijing: People's Posts and Telecommunications Publishing House, 2016: 16-58.

[21] FU J H. Design and implementation of Tourism Map APP based on Android Mobile phone [J]. Information & Communications, 2016(10): 104-105.

[22] LV J D, WANG J T. Design and implementation of Human Heart rate Monitoring system based on single Chip Microcomputer [J]. Electronic Component and Information Technology, 2018(8): 26-28.



SUN Zhi-gang, born in 1996, postgraduate, is a member of China Computer Federation. His main research interests include loose particle detection technology for sealed electronic components and electronic equipment.



JIANG Ai-ping, born in 1962, Ph. D, professor. Her main research interests include digital image processing and computer vision.